Restification

Problem Statement

Many science researchers and power users would like a simple way to acquire large amounts of data via scripts, often recursing through anonymous FTP. Some of these scripters also use filename conventions to select the files they actually want to transfer. However, the unpredictability of some filenames (e.g., due to inclusion of production time stamps) thwarts this.

Proposed Solution:

Restification in this context refers to using the Common Metadata Repository as a resolver for file-level URLs. A convention for URLs would be developed, e.g.,

http://earthdata.nasa.gov/data/<collectionEntryID>/<dataBeginDate>/<collectionEntryID>.<dataBeginDate>.<file_ext>

The HTTP request would be intercepted by CMR, which would parse the URL, based on the convention, query the CMR database for the granule, and return to the client the actual URL to access the data.

Key Benefits:

(1) Data acquisition simplicity

A user would this be able to simply de-reference a URL such as http://earthdata.nasa.gov/data/MOD08_D3.006/20100324/MOD08_D3.006.20100324.hdf in order to get the MODIS L3 Daily Atmosphere data file for 24 March, 2010. CMR would execute the query against granules, and return the contents of the DataAccessURL. No searching would appear to be needed on the user's part, though CMR would be doing a search behind the curtain. This may compensate scripting users for the loss of the anonymous FTP capability.

(2) Data granule landing page option

Just as datasets are said to have a landing page, i.e., a definitive location where one can obtain all the pertinent information about that dataset, so too could a granule have a landing page: if a user appended a .html prefix instead of (or onto?) the actual data extension, a dynamically constructed web page could be returned. This content negotiation aspect could also be extended to granule-level browse (with, say, a .png extension), ISO metadata (.iso),

(3) Data referencing at the file level

While DOI-based data citations are becoming the norm in science papers, referencing the exact files used is still problematic.This could allow researchers to be more precise in describing the data used, though we may not be able to guarantee permanence in quite the same manner as is done for DOIs. (However, note that as a by product, the URL string itself contains useful semantic information providing users with a pretty good lead on how to find any replacement that might exist.

(4) Data distribution routing option

In some cases, it is beneficial to distribute data from alternate repositories (such as to avoid cloud egress charges). The CMR resolver could make decisions among alternatives as to which of alternate URLs should be returned to the user.

(5) Auto-upgrade option

In cases where a user dereferences an obsolete version of data, CMR could substitute a URL for a later data collection. (We would need to communicate this to the user.)

(6) Earthdata branding

All granules being dereferenced this way would have nasa.gov in the REST request URL.

(7) Service options

Subsets could be requested by using w10n-like syntax, converted to an OPeNDAP request.

Reformatting could be requested by changing the filename extension at the end.

Design Considerations

The design of the URL convention is likely the most interesting aspect, and most likely to generate heated discussion. Note that different types of data may need different conventions, particularly in the date-time area. Also, the decision on what unique data collection identifier to use. DOIs

would be ideal, but are inconsistently populated, compared to DIF EntryIDs.

One way to build in some flexibility is to tag the semantic info in the URL with a letter or short string, e.g.,:

doi = DOI

eid = EntryID

bdt = begin date

btm = begin time

In this case, we could begin with entryID as the key to find the right data collection, but switch to DOI support later when search-by-DOI works without "breaking" the URL convention. The MODIS example above would now become:

http://earthdata.nasa.gov/data/eid/MOD08_D3_6/bdt/20100324/MOD08_D3.006.20100324.hdf

or even

http://earthdata.nasa.gov/data/doi/10.5067/MODIS/MOD08_D3.006/bdt/20100324/MOD08_D3.006.20100324.hdf

(Note: the embedded slashes in the DOI do complicate things a bit, but there are ways around this...)

Key Questions

1. Does this kind of redirection work with Earthdata login?
2. Does this kind of redirection work with S3 temporary URLs?

Minimum Viable Prototype

The MVP would need:

1. a URL convention
2. interception of URLs
3. resolution to the data access URL and redirect sent back to client
4. ability to turn the feature on for a given dataset
5. how-to description in the Developer's Portal